mgr inż. Daniel Czyczyn-Egird
Studia doktoranckie: VI semestr

Koszalin, 01.09.2017 r.

Koncepcja przygotowywanej rozprawy doktorskiej

„Predykcja defektów oprogramowania przy użyciu wybranych technik eksploracji danych"

Kontekst zagadnienia

Wykorzystanie modeli klasyfikacyjnych do predykcji defektów oprogramowania pozwala na oszacowanie jakości systemu informatycznego przez określenie liczby znajdujących się w nim defektów jak i wskazanie elementów systemu, które są obarczone ich największą liczbą.

Poprzez defekt można rozumieć pewien niepożądany stan przypadkowy, który powoduje, że jednostka systemu nie działa w sposób wymagany. Jest to jedna z definicji defektu określona normą 982.2 IEEE/ANSI.

Dysponując informacjami o rozmieszczeniu defektów można efektywniej wykorzystać zasoby przeznaczone na testowanie aplikacji, zwiększając zaangażowanie w obszarach systemu wskazanych przy użyciu zaimplementowanego modelu jako te, które mogą zawierać większą liczbę defektów oraz zmniejszając zaangażowanie w obszarach systemu, które zostały oznaczone jako prawdopodobnie wolne od błędów.

W badaniach empirycznych prowadzonych w pracach przez Boehma[1] i Weyuker[2] obwiązuje zasada 80:20, która może mieć również zastosowanie w procesie predykcji defektów oprogramowania – niewielka część kodu źródłowego oprogramowania (około 20%) jest odpowiedzialna za większość defektów (około 80%). Istnieją także inne prace (np. artykuł Weyuker[3]) w których pokazano, ze można w ten sposób zidentyfikować ponad 80% defektów posiłkując się zaledwie 20% kodu źródłowego analizowanego systemu.

Tym samym korzystając z takiego modelu, można spróbować zredukować proces testowania do około 20% kodu źródłowego i mimo tego znaleźć w przybliżeniu 80% defektów ukrytych w testowanym projekcie informatycznym.

Sformułowanie problemu

Przyjmuje się, że jest prowadzony proces wytwarzania oprogramowania dedykowanego dla szerszego grona odbiorców, które z racji dużego rozmiaru projektu może być podatne na wystąpienie w nim szeregu defektów spowodowanych nieumyślnymi błędami programistów

---

[1] Barry W. Boehm, Philip N. Papaccio, "Understanding and controlling software costs", IEEE Transactions on Software Engineering, 14:1462–1477, Październik 1988

[2] Elaine J. Weyuker, Thomas J. Ostrand, Robert M. Bell, "Do too many cooks spoil the broth? Using the number of developers to enhance defect prediction models", Empirical Software Engineering, 13(5):539–559, 2008

[3] Elaine J. Weyuker, Thomas J. Ostrand, Robert M. Bell, "Using developer information as a factor for fault prediction", PROMISE '07: Proceedings of the Third International Workshop on Predictor Models in Software Engineering, Washington, DC, USA, 2007. IEEE Computer Society

implementujących system. Dostępny jest kod źródłowy tworzonego oprogramowania oraz system śledzenia błędów używany w projekcie.

Oczekuje się, że w oparciu o kod źródłowy oraz dane historyczne dotyczące wykrytych wcześniej błędów, pozyskane zostaną konkretne informacje mówiące o możliwych ukrytych problemach, które nie zostały jeszcze zdiagnozowane.

Poszukiwane są modele predykcji oparte na wartościach atrybutów otrzymanych przy użyciu różnych metryk inżynierii oprogramowania. Poprzez zastosowanie tych modeli oczekiwanym rezultatem będzie oszacowanie z jak największą dokładnością liczby defektów oraz obszarów ich występowania w oprogramowaniu. Skutkiem tego powinno być efektywniejsze wykorzystanie zasobów przeznaczonych na testowanie aplikacji. Jednocześnie powinna istnieć możliwość zastosowania owych modeli w projektach informatycznych bez względu na zastosowane w nich języki programowania (pod warunkiem, że dany język programowania wywodzi się z nurtu programowania obiektowego).

Cel rozprawy

Opracowanie oraz zweryfikowanie zestawu modeli klasyfikacyjnych służących do predykcji defektów oprogramowania w projektach programistycznych opartych na wzorcu programowania obiektowego. Wynik klasyfikacji opisujący predykcję możliwych defektów oprogramowania w oparciu o dane wejściowe przyczyni się do zapewnienia lub utrzymania jakości tworzonego oprogramowania.

Teza rozprawy

Hybrydowe klasyfikatory pozwalają na efektywniejszą predykcję defektów oprogramowania w projektach informatycznych tworzonych według paradygmatu programowania obiektowego w stosunku do typowo stosowanych modeli klasyfikacyjnych zapewniając tym samym utrzymanie lub poprawę jakości tworzonego oprogramowania.

Koncepcja proponowanego rozwiązania

1. Zebrać do analizy kilkanaście projektów o otwartej dostępności do kodu oraz posiadających system kontroli wersji z opcją raportowania błędów.
2. Określić atrybuty brane pod uwagę w procesie klasyfikacji.
3. Wyznaczyć dla każdego projektu wartości atrybutów za pomocą metryk produktu oraz procesu, gdzie metryki produktu to metryki mierzalne typu ilość linii kodu, klasy, procedury itp., a metryka procesu to np. stabilność kodu (sprawdzona np. poprzez liczbę wykonanych modyfikacji w systemie wersjonowania itp.). Zebranie wartości dla poszczególnych atrybutów przy użyciu wybranych metryk będzie dotyczyło kodów źródłowych, systemów kontroli wersji (np. SVN), systemów zarządzania projektem i śledzenia błędów (np. JIRA, Fossil, Github).
4. Zaprojektować i zaimplementować hybrydowe modele klasyfikacyjne i sprawdzić ich jakość w zastosowaniu do predykcji defektów oprogramowania.
5. Wybrać najlepsze z opracowanych modeli i podjąć próbę ich wdrożenia do wybranego komercyjnego projektu informatycznego.
6. Opierać pracę badawczą na autorskim oprogramowaniu, starając się unikać gotowych rozwiązań, na rzecz autorskich implementacji poszczególnych technik akwizycji danych oraz technik eksploracji danych.

Postępy w pracy doktorskiej

1. Prace zakończone:
   - poszerzenie swojej wiedzy z zakresów obejmujących takie aspekty jak: predykcja defektów oprogramowania oraz techniki eksploracji danych w ujęciu oceny jakości oprogramowania,
   - zaproponowanie docelowego tematu pracy doktorskiej,
   - zebranie i przygotowanie materiałów pod publikacje z dziedziny badanego zagadnienia (liczba publikacji: 6),
   - publikacja dotycząca przykładu na szacowanie informacji z użyciem metody eksploracji danych, konkretnie regresji liniowej do predykcji kaloryczności wyrobów czekoladowych na podstawie atrybutów jakimi były wartości odżywcze,
   - publikacje dotyczące zbadania popularności wzorców projektowych stosowanych przez programistów w oparciu o portal typu Q&A (stackoverflow.com) z użyciem autorskiego narzędzia do akwizycji danych. Została zaprojektowana komunikacja z API serwisu za pomocą formatu wymiany danych – JSON,
   - publikacje dotyczące predykcji ataków sieciowych typu DDoS za pomocą technik eksploracji danych. Konkretne techniki, które zostały zastosowane to: naiwny klasyfikator bayesowski oraz algorytm *k*-najbliższych sąsiadów,
   - aktywny udział w konferencjach naukowych (liczba konferencji 5, w tym jedna zagraniczna),
   - przygotowanie wstępnej wersji bibliografii powiązanej z tematem pracy doktorskiej,
   - określenie i przygotowanie oprogramowania, zaopatrzenie się w niezbędne urządzenia i licencje niezbędne do stworzenia komputerowych symulacji wspomagających prowadzone prace badawcze.

2. Prace w trakcie realizacji
   - kontynuowanie prac nad stworzeniem oprogramowania do akwizycji danych wymaganych do pracy badawczej,
   - dalsze prace związane z aktualizacją pozycji literatury tematycznej,
   - kompletowanie metryk oprogramowania do zebrania wartości konkretnych atrybutów,
   - rozpoczęcie prac nad oprogramowaniem do predykcji defektów oprogramowania wykorzystujących wybrane techniki eksploracji danych,
   - budowa diagramów obrazujących wybrane metryki produktu i procesu.

Szacunkowy stan zaawansowania postępu pracy: około 50 %.

Przewidywany termin zamknięcia przewodu doktorskiego: rok akademicki 2019/2020.

Spis publikacji

1. Daniel Czyczyn-Egird, *Eksploracja danych na podstawie szacowania informacji z wykorzystaniem regresji liniowej*, w: Modele inżynierii teleinformatyki 10, Wydawnictwo Uczelniane Politechniki Koszalińskiej, ISBN 978-83-7365-365-8, Koszalin 2015
(Liczba punktów: 4 pkt.)

2. Daniel Czyczyn-Egird, Rafał Wojszczyk, *Determining the popularity of design patterns used by programmers based on the analysis of questions and answers on stackoverflow.com Social Network*, w: Communications in Computer and Information Science, **Springer International Publishing**, ISBN 9783319392066, Strony: 421-433, Berlin 2016
(Liczba punktów: 15 pkt.; mój wkład procentowy do publikacji: 50%)

3. Daniel Czyczyn-Egird, Rafał Wojszczyk, *Zastosowanie technik eksploracji danych na przykładzie badania popularności wzorców projektowych w serwisie społecznościowym Stackoverflow.com*, w: Zeszyty Naukowe Wydziału Elektroniki I Informatyki Politechniki Koszalińskiej nr 10, Wydawnictwo Uczelniane Politechniki Koszalińskiej, ISBN 978-83-7365-443-3, Strony: 81-94, Koszalin 2016
(Liczba punktów: 3 pkt.; mój wkład procentowy do publikacji: 50%)

4. Daniel Czyczyn-Egird, Rafał Wojszczyk, *The effectiveness of data mining techniques in the detection of DDoS attacks*, w: Advances in Intelligent Systems and Computing, **Springer International Publishing**, ISBN 978-3-319-62410-5, Strony: 53 – 60, Berlin 2017
(Liczba punktów: 15 pkt.; mój wkład procentowy do publikacji: 50%)

5. Daniel Czyczyn-Egird, Rafał Wojszczyk, *Predykcja ataków DDoS za pomocą technik eksploracji danych, w*: Zeszyty Naukowe Wydziału Elektroniki I Informatyki Politechniki Koszalińskiej, Wydawnictwo Uczelniane Politechniki Koszalińskiej, ISBN 978-83-7365-443-3, Koszalin 2017 – *przyjęte do druku*
(Liczba punktów: 3 pkt.; mój wkład procentowy do publikacji: 50%)

6. Daniel Czyczyn-Egird, Rafał Wojszczyk, *DDoS Attacks Prediction in a Simulation Environment by means of Data Mining Techniques*, w: STUDIA INFORMATICA, Politechnika Śląska, Vol 38, No 3,Strony: 17 – 31, ISSN 1642-0489, Gliwice 2017
(Liczba punktów: 9 pkt.; mój wkład procentowy do publikacji: 50%)

Sumaryczna liczba punktów z publikacji: 49 pkt.

Sumaryczna liczba punktów z publikacji po podziale na autorów według ich wkładu: 26,5 pkt.

Aktywność naukowa

Prezentacje referatów na:

1. XII Krajowa Konferencja Studentów i Młodych Pracowników Nauki, Unieście 2015.
2. XIII Krajowa Konferencja Studentów i Młodych Pracowników Nauki, Szczecin 2016.
3. 23rd International Science Conference on Computer Networks CN2016, Brunów2016.
4. XIV Krajowa Konferencja Studentów i Młodych Pracowników Nauki, Mielno 2017.
5. 14th International Conference on Distributed Computing and Artificial Intelligence (DCAI'17), Polytechnic of Porto, Portugalia 2017.